HW10: Conceptual exercises

Ellen Hsieh

```
library(tidyverse)
library(tidymodels)
library(magrittr)
library(cluster)

options(digits = 3)
set.seed(124)
```

Simulate your own clusters

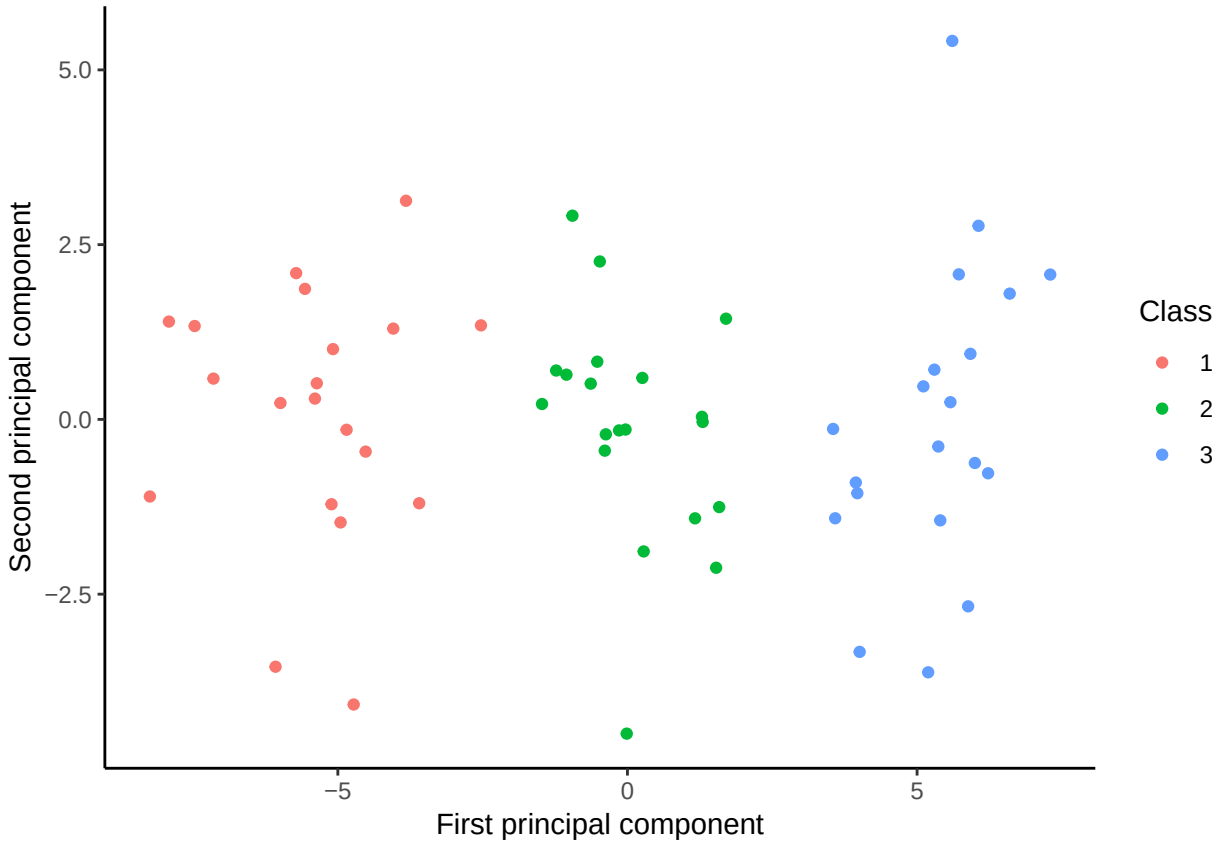
1.

```
X <- rbind(matrix(rnorm(20*50, mean = 0), nrow = 20),
           matrix(rnorm(20*50, mean=0.7), nrow = 20),
           matrix(rnorm(20*50, mean=1.4), nrow = 20))

data <- as_tibble(X) %>%
  mutate(class = factor(c(rep(1, 20),rep(2, 20), rep(3, 20))))
```

```
## Warning: `as_tibble.matrix()` requires a matrix with column names or a `.name_repair` argument. Using
## This warning is displayed once per session.
```

2.

```
X.pca = prcomp(X, scale = FALSE)

augment(X.pca, data = data) %>%
  ggplot(aes(.fittedPC1,.fittedPC2, color = class)) +
  geom_point() + theme_classic() +
  labs(x = "First principal component",
       y = "Second principal component",
       color = "Class")
```



3.

```
res = kmeans(X, centers = 3)
true_class = c(rep(1,20), rep(2,20), rep(3,20))
table(res$cluster, true_class)
```

```
##    true_class
##      1  2  3
##   1  1 20  0
##   2 19  0  0
##   3  0  0 20
```

The result of clustering is nearly perfect.

4.

```
res = kmeans(X, centers = 2)
true = c(rep(1,20), rep(2,20), rep(3,20))
table(res$cluster, true_class)
```

```
##    true_class
##      1  2  3
##   1 20 13  0
##   2  0  7 20
```

The middle class is forced to a wrong class. The extreme classes are classified correctly.

5.

```
res = kmeans(X, centers = 4)
true = c(rep(1,20), rep(2,20), rep(3,20))
table(res$cluster, true_class)
```

```
##    true_class
##      1  2  3
##   1 10  0  0
##   2 10  0  0
##   3  0 20  0
##   4  0  0 20
```

One of the classes is splitt into 2 classes.

6.

```
res <- kmeans(augment(X.pca, data = data) %>% select(.fittedPC1, .fittedPC2), 3)
true = c(rep(1,20), rep(2,20), rep(3,20))
table(res$cluster, true_class)
```

```
##    true_class
##      1  2  3
##   1 19  0  0
##   2  1 20  0
##   3  0  0 20
```

Compared to raw data, the result of clustering is slightly imporved.

7.

```
res = kmeans(scale(X), centers = 3)
true = c(rep(1,20), rep(2,20), rep(3,20))
table(res$cluster, true_class)
```

```
##    true_class
##      1  2  3
##   1  0  0 20
##   2 19  0  0
##   3  1 20  0
```

Scaling of the observations leads to a worse outcome of clustering.

Dissimilarity measures

```
USArrests <- USArrests %>%
  as.matrix()

USArrests_scaled <- t(scale(t(USArrests)))

euclidean_dist <- as.matrix(dist(USArrests_scaled)^2)
euclidean_dist <- euclidean_dist[lower.tri(euclidean_dist)]

corr_dist <- cor(t(USArrests_scaled))
corr_dist <- corr_dist[lower.tri(corr_dist)]
```

```r
prop_dist <- 1 - corr_dist

dist_df = cbind(euclidean_dist, prop_dist) %>%
  as.data.frame()

ggplot(dist_df, aes(euclidean_dist, prop_dist)) +
       geom_point() + theme_classic() +
       labs(title = "Relationship of dissimilarity measures",
             x = "Euclidean distance",
             y = "Correlation distance")
```